UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/118,572	07/17/1998	KARL J. WOOD	PHB34169US	9151

24737 7590 06/25/2008

PHILIPS INTELLECTUAL PROPERTY & STANDARDS
P.O. BOX 3001
BRIARCLIFF MANOR, NY 10510

EXAMINER	
YANG, RYAN R	
ART UNIT	PAPER NUMBER
2628	

MAIL DATE	DELIVERY MODE
06/25/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte KARL J. WOOD, CORNELIS W. A. M. VAN OVERVELD, HENDRIK DIJKSTRA, and DOUGLAS R. M. PATTERSON

Appeal 2008-0657
Application 09/118,572
Technology Center 2600

Decided: June 25, 2008

Before TERRY J. OWENS, HOWARD B. BLANKENSHIP, and SCOTT R. BOALICK, *Administrative Patent Judges*.

OWENS, *Administrative Patent Judge*.

DECISION ON APPEAL

The Appellants appeal from a rejection of claims 1-5, 7 and 9. Claims 6, 8, 10 and 11, which are all of the other pending claims, stand objected to but allowable if rewritten in independent form.

THE INVENTION

The Appellants claim a texture mapping apparatus. Claim 1 is illustrative:

1. Apparatus for texture mapping in a computer graphics system, using a predetermined set of standardized textures, the apparatus having an input to receive via a network identifying data identifying one of the set of standardized textures, and means for processing the data to generate output texels of the identified texture, wherein each texture of the standardized set is a procedural texture, the identifying data comprises one or a sequence of program commands, the execution of which will result in the generation of a respective procedural texture, with the means for processing data comprising a processor operable to implement all such input program commands or sequences of input program commands as required to generate the procedural textures of the standardized set.

THE REFERENCE

John Rhoades et al., "Real-Time Procedural Textures", *Proc. 1992 Symp. in Interactive 3D Graphics* 95-100, June 1992.

THE REJECTION

Claims 1-5, 7, and 9 stand rejected under 35 U.S.C. § 102(b) over Rhoades.

OPINION

We reverse the Examiner's rejection. We need to address only the sole independent claim, i.e., claim 1.

Rhoades discloses software for interactively creating, editing and displaying user-defined antialiased procedural textures for use in real time graphics applications (abstract). The procedural textures are implemented using an assembly language-like instruction set called T-codes, the interpretation of which by graphics processors produces an image generation

controller (IGC) command instruction stream that is routed to renderers (128x128 arrays of bit-serial pixel processors capable of general arithmetic and logical operations) for single instruction multiple data (SIMD) execution (p. 96, left col., lines 12-18, 27-31, right col., ll. 14-15, 21-23). A "texture editor displays the T-code instructions of a selected procedural texture in a text window" (p. 98, left col., ll. 14-15). "T-code instructions can be added, rearranged, and deleted, producing a new program. Then with a couple of commands, the user can save the updated texture program and reload it into the texture editor for immediate display" (p. 98, left col., ll. 26-30). Rhoades teaches that "[a]dding a new T-code to our system is a straightforward task. Besides coding and testing of the T-code subroutine in C, the programmer needs only to update the T-code assembler parse table and the T-code subroutine dispatch table" (p. 96, right col., ll. 36-40). The system used to implement the T-codes includes graphics processors, renderers, frame buffers and a workstation host communicating over a shared ring network (p. 96, left col., ll. 20-22; fig. 1).

The Examiner has the initial burden of establishing a prima facie case of anticipation by pointing out where all of the claim limitations appear in a single reference. *See In re Spada*, 911 F.2d 705, 708 (Fed. Cir. 1990); *In re King*, 801 F.2d 1324, 1327 (Fed. Cir. 1986).

The Appellants' claim 1 requires "an input to receive via a network identifying data".

The Examiner states that "Examiner considers Rhodes' [sic] ring network meets the limitation" (Ans. 7).

Rhoades' ring network consists of graphics processors, renderers, frame buffers and a workstation host (p. 96, left col., ll. 20-22; fig. 1).

Rhoades' T-code data is not received via that network but, rather, is input to that network. The ring network's graphics processors then interpret the T-code program and produce IGC command instruction streams that are routed to renderers for SIMD execution (p. 96, left col., ll. 18-23, right col., ll. 36-40).

The Appellants' claim 1 requires "identifying data identifying one of the set of standardized textures".

The Examiner argues that Rhoades' "updated texture program becomes a new predetermined set of standardized textures" (Ans. 6-7).

The ordinary meaning of "standardize" is "**1**: to compare with a standard **2**: to bring into conformity with a standard".[1] Rhoades' procedural textures are user defined and can be added by updating a T-code assembler parse table and a T-code subroutine dispatch table (abstract; p. 96, right col. ll. 36-40). The Examiner has not explained how those user-defined T-codes are compared with a standard, brought into conformity with a standard, or otherwise rendered standardized. The Examiner's mere assertion that Rhoades' T-codes are standardized does not establish that they indeed are standardized.

The Appellants' claim 1 requires "the identifying data comprises one or a sequence of program commands, the execution of which will result in the generation of a respective procedural texture".

The Examiner argues: "Examiner considers the IGC command instruction stream is a sequence of command and is sent to a Renderer (page 96, column 2, line[s] 20-23). Since a Renderer is a processor, it meets the limitation" (Ans. 7).

[1] *Webster's New Collegiate Dictionary* 1133 (G. & C. Merriam 1973).

Rhoades' renderers receive IGC command instruction streams (p. 96, ll. 21-23). The execution of those instructions by the renderers, however, does not generate a procedural texture. The renderers are pixel processors that rasterize polygons (p. 96, right col., l. 27 – left col., l. 10). Rhoades' program commands for generating procedural textures are the T-code instruction set, and those instructions are input to graphics processors that interpret the instructions and produce the IGC command instruction streams routed to the renderers (p. 96, right col., ll. 13-23).

For the above reasons we find that the Examiner has not established a prima facie case of anticipation of the Appellants' claimed invention.

DECISION

The rejection of claims 1-5, 7 and 9 under 35 U.S.C. § 102(b) over Rhoades is reversed.

REVERSED

PL initials:
sld

PHILIPS INTELLECTUAL PROPERTY & STANDARDS
P.O. BOX 3001
BRIARCLIFF MANOR, NY 10510